ADVANCING AI PERFORMANCE
WITH INTEL® XEON® SCALABLE SYSTEMS

BANU NAGASUNDARAM & VIKRAM SALETORE
ADVANCING AI PERFORMANCE WITH INTEL® XEON® SCALABLE

TIME TO SOLUTION FOR PRODUCTION AI

WORKLOAD & SCALABILITY FLEXIBILITY

MAXIMIZE PERFORMANCE USE OPTIMIZED SW

JOURNEY TO PRODUCTION AI

DEEP LEARNING IN DATA CENTERS

INTEL AI FOCUS PILLARS
INTERSECTION OF DATA AND COMPUTE GROWTH

Daily by 2020

- **Average Internet User**: 1.5 GB
- **Autonomous Vehicle**: 4 TB
- **Connected Airplane**: 5 TB
- **Smart Factory**: 1 PB
- **Cloud Video Provider**: 750 PB

**Business Insights**

**Operational Insights**

**Security Insights**

Source: Amalgamation of analyst data and Intel analysis.
Data Analytics needs AI

Self-Learning and Completely Automated Enterprise
Simulation-Driven Analysis and Decision-Making
Foresight
What Will Happen, When, and Why

Operational Analytics
Diagnostic Analytics
Descriptive Analytics

Advanced Analytics
Predictive Analytics
Prescriptive Analytics
Cognitive Analytics

Insight
What Happened and Why
Hindsight
What Happened

Data Deluge
Compute Breakthrough
Innovation Surge

Today
Emerging
# AI WITH INTEL

**CONSUMER**
- Recommend products to customers

**HEALTH**
- Computational screen drug candidates

**FINANCE**
- Time-based pattern detection

**RETAIL**
- Image recognition

**GOVERNMENT**
- Furthered the search for lunar ice

**ENERGY**
- Detect and classify corrosion levels

**TRANSPORT**
- Issuing traffic tickets for violations recorded by cameras

**INDUSTRIAL**
- Automate increased inspection frequency

**OTHER**
- Silicon packaging inspection

**INSURANCE PROVIDER**
- Increased accuracy

**KYOTO UNIVERSITY**
- Chose Xeon over GPU, decreased time & cost

**US MARKET EXCHANGE**
- 10x reduction in search costs

**JD.COM**
- 4x gain by switching from GPUs to Xeon

**NASA**
- Automating lunar crater detection using a CNN

**LEADING OIL COMPANY**
- Full automation

**SERPRO**
- Full automation

**SOLAR FARM**
- Reduced time to train

**INTEL PROJECTS**
- Significant speedup

---

3. Refer slide 61
4. Refer slide 61
7. Refer slide 62

---

**Source:**
Refer builders.intel.com/ai/solutionslibrary

---

For more information refer builders.intel.com/ai/solutionslibrary

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as CineMark and Moosher, are measured using specific software components and should not be inferred to represent performance of any other software application. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software applications, and methodologies; results are not necessarily indicative of performance or comparison of any other products and may not reflect some or any specific characteristics or features of other products.

For more complete information visit: http://www.intel.com/processor/compare

---

**Note:** Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as CineMark and Moosher, are measured using specific software components and should not be inferred to represent performance of any other software application. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software applications, and methodologies; results are not necessarily indicative of performance or comparison of any other products and may not reflect some or any specific characteristics or features of other products.

For more complete information visit: http://www.intel.com/processor/compare
Deep learning in practice

Innovation Cycle

- Label data: 15%
- Load data: 15%
- Augment data: 23%
- Experiment with topologies: 15%
- Tune hyper-parameters: 15%
- Support inference inputs: 8%
- Document results: 8%

Development Cycle

- Source Data
- Inferencing
- Inference within broader application

Time-to-Solution

- Source Data
- Deploy
- Data Integration & Management
- Data Processing
- AI
- Decision Process
- Broader Application
- Refresh

Production Deployment

Data Store

Time-to-solution is more significant than time-to-train

Source: Intel estimates based on customer data
THE JOURNEY TO PRODUCTION AI

TOTAL TIME TO SOLUTION

Frame Opportunity
- Define the problem
- Assess data needs

Hypotheses
- Identify software model needs
- Categorize existing resources

Data Prep
- Collect data
- Store data
- Extract, transform & load (ETL)
- Label data
- Process data
- Test data

Modeling
- Experiment with models
- Develop a proof of concept
- Train model
- Measure performance
- Evaluate system costs

Deployment
- Integrate into workflow
- Build organizational and system level capabilities to scale

Iteration
- Increase robustness of model with additional data
- Re-training and refine model
- Tie inference to workflow decision making
- Optimize for end-user experience

Evaluation
- Measure utilization
- Explore need for additional processing hardware
- Understand bottlenecks to scaling
- Assess additional revenue generation models

Source: Intel estimates based on customer data
# Intel® AI Portfolio

## Solutions
- Data Scientists
- Technical Services
- Reference Solutions

## Platforms
<table>
<thead>
<tr>
<th>Intel® AI Builders</th>
<th>Intel® Deep Learning System†</th>
<th>Intel® Saffron™ Reasoning</th>
</tr>
</thead>
</table>

## Tools
| Intel® Deep Learning Studio‡ | Intel® Deep Learning Deployment Toolkit† | OpenVINO™ Toolkit | Intel® Movidius™ Software Development Kit (SDK) |

## Frameworks
- TensorFlow
- Caffe
- mxnet
- DL4J
- Spark
- Caffe2
- PyTorch
- CNTK
- others

## Libraries
- Intel® MKL-MKL-DNN, cDNN, DAAL, Intel® Python Distribution, etc.
- Intel® nGraph™ Compiler
- DIRECT OPTIMIZATION
  - CPU Transformer†
  - NNP Transformer‡
  - Other

## Technology
- End-to-End Compute
- Systems & Components

*Alpha available
†Beta available
‡Future

*Other names and brands may be claimed as the property of others.

*All products, computer systems, dates, and figures are preliminary based on current expectations, and are subject to change without notice.*
Compute

Multi-purpose to purpose-built compute for AI workloads from cloud to device

General AI

Foundation for AI

Training

Intensive Training +

Mainstream Training

Inference

Targeted Inference +

Mainstream Inference

Accelerated Deep Learning

Intel Nervana (IP)

Intel Xeon Platinum inside

Intel Graphics

Infer Graphics

Infer Movidius

AIDC
DEEP LEARNING IN DATA CENTERS
DATA CENTERS - FLEXIBILITY AND SCALABILITY

Re-provision resources when AI developers do not need system access

Achieving PUE close to 1
Performance at scale
Efficient asset utilization

Data Center Priorities

Provide access to multiple nodes through scalable performance when compute needs come in

Source: Facebook, Inc. Paper – Applied Machine Learning at Facebook: A Datacenter Infrastructure Perspective
The Register® https://www.theregister.co.uk/2017/05/22/cloud_providers_ai_researchers/ AWS Pricing Analysis – Screen Capture 23 May 2017
**Built-in ROI with Intel® Xeon® Clusters**

Workload Flexibility with Multi-Purpose CPU

Inference throughput needed per day

<table>
<thead>
<tr>
<th>100 Million images/day</th>
<th>1 Billion images/day</th>
<th>2 Billion images/day</th>
<th>4 Billion images/day</th>
</tr>
</thead>
<tbody>
<tr>
<td>0%</td>
<td>Underutilized CPUs</td>
<td>should be used for</td>
<td>inference</td>
</tr>
<tr>
<td>100%</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Utilization estimated on 64-node cluster with estimated performance on Caffe Resnet 50 inference throughput with 25 Intel® Xeon® Scalable Platinum 8180 Processor. User configuration results. Performance measurements were obtained prior to implementation of recent software patches and firmware updates intended to address exploits referred to as "Spectre" and "Meltdown." Implementation of the recent software patches and firmware updates may or may not optimize the same features for end-user experience for organizations that are not unique to Intel microprocessors. These optimizations include GPR 128, register file, and DDR4. Performance tests are not intended to be comparable to the results of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations, in this product, are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations, and features. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchase, including the performance of that product when combined with other products. For more complete information visit: [http://www.intel.com/content/www/us/en/performance.html](http://www.intel.com/content/www/us/en/performance.html).

Utilization estimated on 64-node cluster with estimated performance on Caffe Resnet 50 inference throughput with 25 Intel® Xeon® Scalable Platinum 8180 Processor. User configuration results. Performance measurements were obtained prior to implementation of recent software patches and firmware updates intended to address exploits referred to as "Spectre" and "Meltdown." Implementation of the recent software patches and firmware updates may or may not optimize the same features for end-user experience for organizations that are not unique to Intel microprocessors. These optimizations include GPR 128, register file, and DDR4. Performance tests are not intended to be comparable to the results of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations, in this product, are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations, and features. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchase, including the performance of that product when combined with other products. For more complete information visit: [http://www.intel.com/content/www/us/en/performance.html](http://www.intel.com/content/www/us/en/performance.html).

Utilization estimated on 64-node cluster with estimated performance on Caffe Resnet 50 inference throughput with 25 Intel® Xeon® Scalable Platinum 8180 Processor. User configuration results. Performance measurements were obtained prior to implementation of recent software patches and firmware updates intended to address exploits referred to as "Spectre" and "Meltdown." Implementation of the recent software patches and firmware updates may or may not optimize the same features for end-user experience for organizations that are not unique to Intel microprocessors. These optimizations include GPR 128, register file, and DDR4. Performance tests are not intended to be comparable to the results of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations, in this product, are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations, and features. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchase, including the performance of that product when combined with other products. For more complete information visit: [http://www.intel.com/content/www/us/en/performance.html](http://www.intel.com/content/www/us/en/performance.html).
The abundance of readily-available CPU capacity makes it a useful platform for both training and inference. This is especially true during the off-peak portions of the diurnal cycle where CPU resources would otherwise sit idle.

<table>
<thead>
<tr>
<th>Services</th>
<th>Ranking Algorithm</th>
<th>Photo Tagging</th>
<th>Photo Text Generation</th>
<th>Search</th>
<th>Language translation</th>
<th>Spam Flagging</th>
<th>Speech Recognition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Models</td>
<td>MLP</td>
<td>SVM,CNN</td>
<td>CNN</td>
<td>MLP</td>
<td>RNN</td>
<td>GBDT</td>
<td>RNN</td>
</tr>
<tr>
<td>Inference Resource</td>
<td>CPU</td>
<td>CPU</td>
<td>CPU</td>
<td>CPU</td>
<td>CPU</td>
<td>CPU</td>
<td>CPU</td>
</tr>
<tr>
<td>Training Resource</td>
<td>CPU</td>
<td>GPU &amp; CPU</td>
<td>GPU</td>
<td>Depends</td>
<td>GPU</td>
<td>CPU</td>
<td>GPU</td>
</tr>
<tr>
<td>Training Frequency</td>
<td>Daily</td>
<td>Every N photos</td>
<td>Multi-Monthly</td>
<td>Hourly</td>
<td>Weekly</td>
<td>Sub-Daily</td>
<td>Weekly</td>
</tr>
<tr>
<td>Training Duration</td>
<td>Many Hours</td>
<td>Few Seconds</td>
<td>Many Hours</td>
<td>Few Hours</td>
<td>Days</td>
<td>Few Hours</td>
<td>Many Hours</td>
</tr>
</tbody>
</table>

Performance estimates were obtained prior to implementation of recent software patches and firmware updates intended to address exploits referred to as "Spectre" and "Meltdown." Implementation of these updates may make these results inapplicable to your device or system. Optimization Notice: Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as DiMark and MichiMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of these factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit: http://www.intel.com/performance. Source: Intel measured or estimated as of November 2017.
14X HIGHER INFECTION PERFORMANCE ON INTEL® XEON® SCALABLE PROCESSORS ON NEURAL MACHINE TRANSLATION

Performance measurements were obtained prior to implementation of recent software patches and firmware updates intended to address exploits referred to as “Spectre” and “Meltdown.” Implementation of these updates may make these results inapplicable to your device or system. Optimization Notice: Intel’s compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit http://www.intel.com/content/www/us/en/ark/products.html. Source: Tests as of May 2018.
For ImageNet inference, Intel submitted the best result in both cost and latency. Using an Intel optimized version of Caffe on high performance AWS instances, they reduced per image latency to 9.96 milliseconds and processed 10,000 images for $0.02.
## AI FOCUS PILLARS

<table>
<thead>
<tr>
<th>MAXIMIZE PERFORMANCE</th>
<th>INNOVATE HARDWARE SOLUTIONS</th>
<th>ACCELERATE DEPLOYMENTS</th>
<th>ECOSYSTEM ENABLEMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Through continuous software optimizations to libraries and frameworks</td>
<td>By architecting innovative solutions to improve underlying hardware capabilities for AI</td>
<td>Speed up customer deployments through turn-key solutions for AI applications</td>
<td>Partner with customers &amp; developers on their AI journey to develop end to end solutions from edge to cloud</td>
</tr>
</tbody>
</table>
MAXIMIZE PERFORMANCE
## Inference Throughput Images/Second on Intel Caffe ResNet50

<table>
<thead>
<tr>
<th>Date</th>
<th>Configuration</th>
<th>Images/Second</th>
</tr>
</thead>
<tbody>
<tr>
<td>July 11, 2017</td>
<td>FP32 BS = 64, 2S Intel® Xeon® Scalable Platinum 8180 Processor</td>
<td>131</td>
</tr>
<tr>
<td>July 11, 2017</td>
<td>FP32 BS = 64, 2S Intel® Xeon® Scalable Platinum 8180 Processor</td>
<td>226</td>
</tr>
<tr>
<td>December 14, 2017</td>
<td>FP32 BS = 64, 2S Intel® Xeon® Scalable Platinum 8180 Processor</td>
<td>652</td>
</tr>
<tr>
<td>January 19, 2018</td>
<td>FP32 BS = 64, 2S Intel® Xeon® Scalable Platinum 8180 Processor</td>
<td>800</td>
</tr>
<tr>
<td>April 3, 2018</td>
<td>Int8 BS = 64, 2S Intel® Xeon® Scalable Platinum 8180 Processor</td>
<td>1110</td>
</tr>
<tr>
<td>May 8, 2018</td>
<td>Int8 BS = 64, 2S Intel® Xeon® Scalable Platinum 8180 Processor</td>
<td>1225</td>
</tr>
</tbody>
</table>

### Performance Improvement

In 10 months since Intel® Xeon® Scalable Processor launch, there has been a **5.4X** performance improvement with software optimizations on Caffe Resnet-50. Performance measurements were obtained prior to implementation of recent software patches and firmware updates intended to address exploits referred to as "Spectre" and "Meltdown." Implementation of these updates may make these results inapplicable to your device or system.

### Optimization Notice

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Software and workloads used in performance tests may have been optimized for performance only on Intel微architecture products. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations, and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit: [http://www.intel.com/performance](http://www.intel.com/performance).

**Source:** Intel measured as of April 2018.
OPTIMIZED SOFTWARE : MKL-DNN LIBRARY

2D & 3D Convolution  2D & 3D Inner Product

Pooling   Normalization

Activation:
- ReLU (Training)
- Tanh, Logistic Regression, Softmax (Inference)

Data Manipulation: Reorder, Sum, Concat
Layer Fusion Example

Optimized software: MKL-DNN Library

Input

Convolution 1x1
Convolution 3x3
Convolution 1x1

SUM

Memory Write
Memory Read

ReLU

Memory Write
Memory Read

Output

Memory Ops reduced

Input

Convolution 1x1
Convolution 3x3
Fused primitive
Convolution 1x1 + SUM + ReLU

Output

Memory Write
Memory Read
Winograd Algorithm to improve matrix multiply performance

**Normal Matrix Multiply**
6 multiplications

\[
Y_0 = i_0 \cdot F_0 + i_1 \cdot F_1 + i_2 \cdot F_2 \\
Y_1 = i_1 \cdot F_1 + i_2 \cdot F_2 + i_3 \cdot F_3
\]

\[
X_0 = (i_0 - i_2) \cdot F_0 \\
X_1 = (i_1 + i_2) \cdot (F_0 + F_1 + F_2) / 2 \\
X_2 = (i_1 - i_3) \cdot F_2 \\
X_3 = (i_2 - i_1) \cdot (F_0 - F_1 + F_2) / 2
\]

**Winograd Matrix Multiply**
4 multiplications

\[
Y_0 = X_0 + X_1 + X_2 \\
Y_1 = X_1 - X_2 - X_3
\]

Compute Operations decreased, memory accesses increased

**Optimized Software:** MKL-DNN Library
USE OPTIMIZED MKL-DNN LIBRARY
+
USE OPTIMIZED FRAMEWORKS
Important to use optimized software frameworks and libraries for best AI workload performance

Example: Load Balancing:

TensorFlow graphs offer opportunities for parallel execution. Threading model

1. `inter_op_parallelism_threads`: Max number of operators that can be executed in parallel
2. `intra_op_parallelism_threads`: Max number of threads to use for executing an operator
3. `OMP_NUM_THREADS`: MKL-DNN equivalent of `intra_op_parallelism_threads`
## Optimized Framework Installation

<table>
<thead>
<tr>
<th>Framework</th>
<th>How to Access Optimized Framework</th>
</tr>
</thead>
<tbody>
<tr>
<td>TensorFlow</td>
<td>Install Intel optimized wheel, see tensorflow.org page for CPU optimization instructions</td>
</tr>
<tr>
<td>MXNet</td>
<td>Intel optimizations in main branch via experimental path, available here</td>
</tr>
<tr>
<td>Caffe2</td>
<td>Will upstream to master branch in Q2</td>
</tr>
<tr>
<td>PaddlePaddle</td>
<td>Paddle Paddle master branch</td>
</tr>
<tr>
<td>PyTorch</td>
<td>Intel optimizations available in this branch</td>
</tr>
<tr>
<td>Caffe</td>
<td>Intel optimized version of Caffe</td>
</tr>
<tr>
<td>CNTK</td>
<td>Will upstream to master branch in Q2</td>
</tr>
</tbody>
</table>
INFEERENCE

USE OPTIMIZED MKL-DNN LIBRARY

USE OPTIMIZED FRAMEWORKS

+ ENABLE MULTIPLE STREAMS
Recommend using multiple framework instances
Each framework instance is pinned to a separate NUMA domain
Each instance processes a separate Inference Stream

Optimizations at run time **without** framework code change

**Inference Example: Multi-Stream for TensorFlow**

For 2S Intel Xeon® Platinum 8170 processor-based systems, sub-socket with 8 inference streams:

- **common_args**: `--model resnet50 --batch_size 64 --data_format NCHW --num_batches 100 --distortions=True --mkl=True --num_warmup_batches 10 --device cpu --data_dir ~/tensorflow/TF_Records --data_name imagenet --display_every 10`
- **WK_HOST**: "hostname"
- **worker_env**: "export OMP_NUM_THREADS=6"
- **inf_args**: "$common_args --num_intra_threads 6 --num_inter_threads 2"

To start 4 inference streams on Socket-1:

- `ssh $WK_HOST; $worker_env; nohup unbuffer numactl -m 1 python tf_cnn_benchmarks.py --forward_only True $inf_args --kmp_affinity="granularity=thread,proclist=[19-25,71-77],explicit,verbose" &`
- `ssh $WK_HOST; $worker_env; nohup unbuffer numactl -m 1 python tf_cnn_benchmarks.py --forward_only True $inf_args --kmp_affinity="granularity=thread,proclist=[39-44,91-96],explicit,verbose" &`
- `ssh $WK_HOST; $worker_env; nohup unbuffer numactl -m 1 python tf_cnn_benchmarks.py --forward_only True $inf_args --kmp_affinity="granularity=thread,proclist=[45-51,96-102],explicit,verbose" &`

To start 4 inference streams on Socket-0:

- `ssh $WK_HOST; $worker_env; nohup unbuffer numactl -m 0 python tf_cnn_benchmarks.py --forward_only True $inf_args --kmp_affinity="granularity=thread,proclist=[0-5,52-57],explicit,verbose" &`
- `ssh $WK_HOST; $worker_env; nohup unbuffer numactl -m 0 python tf_cnn_benchmarks.py --forward_only True $inf_args --kmp_affinity="granularity=thread,proclist=[6-12,58-64],explicit,verbose" &`
- `ssh $WK_HOST; $worker_env; nohup unbuffer numactl -m 0 python tf_cnn_benchmarks.py --forward_only True $inf_args --kmp_affinity="granularity=thread,proclist=[13-18,65-70],explicit,verbose" &`
- `ssh $WK_HOST; $worker_env; nohup unbuffer numactl -m 0 python tf_cnn_benchmarks.py --forward_only True $inf_args --kmp_affinity="granularity=thread,proclist=[19-25,71-77],explicit,verbose" &`
3X Higher Inference Performance

By Using MKL-DNN Libraries + Optimized Framework

On MxNet Amazon* C5 (Intel® Xeon® Scalable Processor) running NMT1(German to English) with and without MKL DNN libraries

5X Higher Inference Performance

By Using Multiple Streams

On MxNet Amazon* C5 (Intel® Xeon® Scalable Processor) running NMT1(German to English) with MKL DNN libraries comparing with and without multiple streams
INTEL® XEON® SCALABLE PROCESSORS PERFORMANCE ON NEURAL MACHINE TRANSLATION

Speedup compared to a baseline of 1.0 without MKL measured in Sentences/second

MxNet Amazon® C5 (Intel® Xeon® Processor)
NMT1(German to English)

Higher is Better

Batch Size = 1 | Batch Size = 2 | Batch Size = 8 | Batch Size = 16 | Batch Size = 32 | Batch Size = 64 | Batch Size = 128
---|---|---|---|---|---|---
c5.18xlarge (Skylake) NO MKL | 2.7 | 9.5 | 11.3 | 11.9 | 11.7 | 10.7 | 10

c5.18xlarge (Skylake)MKL/OMP KMP | 14.7 | 5 | 12.6 | 13.2 | 13 | 12 | 12

c5.18xlarge (Skylake)MKL/OMP KMP/6 workers | 14.7 | 5 | 12.6 | 13.2 | 13 | 12 | 12

Source: Intel measured as of April 2018.

*sockeye https://github.com/awslabs/sockeye

Measured by Intel on AWS instance. Measured as of May 2018 Configurations:34.35. Performance measurements were obtained prior to implementation of recent software patches and firmware updates intended to address exploits referred to as "Spectre" and "Meltdown". Implementation of these updates may make these results inapplicable to your device or system. Optimization hence, Intel’s compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may affect your results. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information, visit: http://www.intel.com/performance

Higher is Better
TRAINING

USE OPTIMIZED MKL-DNN LIBRARY

USE OPTIMIZED FRAMEWORKS

ENABLE MULTIPLE STREAMS

+ SCALEOUT TO MULTIPLE NODES
Each framework instance is pinned to a separate NUMA domain
Each CPU running 1 or more workers/node
Uses optimized MPI library for gradient updates over shared memory
Caffe – Use Optimized Intel® MPI ML Scaling Library (ML-SL)
TensorFlow – Uber horovod MPI Library

Optimizations at run time without framework code change

Distributed Deep Learning Training Across Multiple nodes
Each node running multiple workers/node
Uses optimized MPI library for gradient updates over network fabric
Caffe – Use Optimized Intel® MPI ML Scaling Library (ML-SL)
TensorFlow – Uber horovod MPI Library

For 4-Node 2S 28 Core Intel Xeon® Platinum 8180 processor based cluster with 4 Workers/Node
Total of 16 TensorFlow Workers using horovod MPI Communication Library:

OMP_NUM_THREADS=14

To start Distributed Training:

mpiexec --machinefile <hostfile> -genv -np 16 -ppn 4 -genv OMP_NUM_THREADS $OMP_NUM_THREADS \ 
-genv I_MPI_PIN_DOMAIN 28:compact -genv HOROVD_FUSION_THRESHOLD 134217728 \ 
python <path>/tf_cnn_benchmarks/tf_cnn_benchmarks.py --batch_size=64 --model=resnet50 \ 
--num_inter_threads 2 --num_intra_threads $OMP_NUM_THREADS\ 
--num_batches 100 --display_every 10 --data_format NCHW --optimizer momentum --device cpu --mkl=true \ 
--variable_update horovod --horovod_device cpu --local_parameter_device cpu \ 
--kmp_blocktime=1 --enable_layout_optimizer=TRUE --data_dir=<path-to-TFRecords> \ 
--data_name=<dataset_name>

Where hostfile is the file containing the hostnames, one on each new line
Efficient DL scaling on existing infrastructure

Configuration Details

Performance measurements were obtained prior to implementation of recent software patches and firmware updates intended to address exploits referred to as "Spectre" and "Meltdown." Implementation of these updates may make these results inapplicable to your device or system.

Optimization Notice: Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations, and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

Source: Intel measured as of April 2018.

Best Practices From SURFsara B.V.: https://surfdrive.surf.nl/files/index.php/s/xrEFLPvo7IDRARs
INNOVATE HARDWARE SOLUTIONS
TRAINING ENHANCEMENTS

EMBEDDED ACCELERATION WITH AVX-512

AVX-512 Instructions bring embedded acceleration for AI on Intel® Xeon® Scalable processors

<table>
<thead>
<tr>
<th>Sign</th>
<th>Exponent</th>
<th>Mantissa</th>
</tr>
</thead>
<tbody>
<tr>
<td>31</td>
<td>30</td>
<td>29</td>
</tr>
<tr>
<td>28</td>
<td>27</td>
<td>26</td>
</tr>
<tr>
<td>25</td>
<td>24</td>
<td>23</td>
</tr>
<tr>
<td>22</td>
<td>21</td>
<td>20</td>
</tr>
<tr>
<td>19</td>
<td>18</td>
<td>17</td>
</tr>
<tr>
<td>16</td>
<td>15</td>
<td>14</td>
</tr>
<tr>
<td>13</td>
<td>12</td>
<td>11</td>
</tr>
<tr>
<td>10</td>
<td>09</td>
<td>08</td>
</tr>
<tr>
<td>07</td>
<td>06</td>
<td>05</td>
</tr>
<tr>
<td>04</td>
<td>03</td>
<td>02</td>
</tr>
<tr>
<td>01</td>
<td>00</td>
<td></td>
</tr>
</tbody>
</table>

Typical AVX-512 instruction to perform FP32 convolutions: **vfmad231ps**

INPUT FP32

**vfmad231ps**

OUTPUT FP32
Inference Enhancements

Vector Neural Network Instructions

Low Precision Instructions bring embedded acceleration for AI on Intel® Xeon® Scalable

**vpmaddubsw, vpmaddwd, vpadd**

Current AVX-512 instructions to perform INT8 convolutions:

Future AVX-512 (VNNI) instruction to accelerate INT8 convolutions: **vpdpbusd**

**vpmaddubsw, vpmaddwd, vpaddd → vpdpbusd**
ACCELERATE DEPLOYMENTS
DEPLOYING TENSORFLOW WITH SINGULARITY

- Install Singularity on the Infrastructure Nodes
  - https://singularity.lbl.gov/install-linux - as root/sudo

- Build a TensorFlow Singularity Image: tf_singularity.simg
  - Build an image comprising of
    - Linux OS
    - Optimized TensorFlow*
    - Horovod Communication MPI library
    - TensorFlow Application

- Running Application
  - singularity exec tf_singularity.img \
    python /<path-to-benchmarks>/tf_cnn_benchmarks/tf_cnn_benchmarks.py --model resnet50 \
    --batch_size 64 --data_format NCHW --num_batches 1000 --distortions=True --mkl=True --device cpu \
    --num_intra_threads $OMP_NUM_THREADS --num_inter_threads 2 --kmp_blocktime=0
ECOSYSTEM ENABLEMENT
# AI ECOSYSTEM & RESEARCH

**Establish Thought Leadership, Innovate on Intel Architecture, Synergize Intel Product & Research**

<table>
<thead>
<tr>
<th>Topic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Computational Intelligence</strong></td>
<td>Heterogenous architectures for adaptive and always learning devices with NLP and conversational understanding capabilities and visual applications.</td>
</tr>
<tr>
<td><strong>Experiential Computing</strong></td>
<td>3D scene understanding using DL based analysis of large video databases, Computer Vision in the cloud – enable effective data mining of large collections of Video.</td>
</tr>
<tr>
<td><strong>Approximate Computing</strong></td>
<td>Always-on audio-visual multi-modal interaction, Self configuring audio-visual hierarchical sensing through approximate computing data path.</td>
</tr>
<tr>
<td><strong>Deep Learning Architecture</strong></td>
<td>Deep Learning hardware and software advancements, scaling to very large clusters and new applications.</td>
</tr>
<tr>
<td><strong>Visual Cloud Systems</strong></td>
<td>Large-scale systems research for scaling out visual applications and data, large-scale video analysis.</td>
</tr>
<tr>
<td><strong>Security in Deep Learning</strong></td>
<td>Built-in safety mechanisms for wide spread mission critical use, ascertaining the confidence and removing anomalous and out of distribution samples in autonomous driving, medicine, security.</td>
</tr>
<tr>
<td><strong>Brain Science</strong></td>
<td>Create an instrument to connect human behavior to brain function, toolkits for the analysis of brain function, Real-time cloud services for neuroimaging analysis Applying neuroscientific insights to AI.</td>
</tr>
</tbody>
</table>

---

**Intel Invests $1+ Billion in the AI Ecosystem to Fuel Adoption and Product Innovation**

- Intel Invests $1+ Billion in the AI Ecosystem to Fuel Adoption and Product Innovation
- 100+ University engagements

---

ADVANCING AI PERFORMANCE WITH INTEL® XEON® SCALABLE

- **Time to Solution for Production AI**
- **Workload Flexibility & Scalability**
- **Maximize Performance Use Optimized SW**

Journey to Production AI

Deep Learning in Data Centers

Intel AI Focus Pillars
NOTICES & DISCLAIMERS

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

Intel technologies’ features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at intel.com, or from the OEM or retailer. No computer system can be absolutely secure.

Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. Consult other sources of information to evaluate performance as you consider your purchase. For more complete information about performance and benchmark results, visit http://www.intel.com/performance.

Cost reduction scenarios described are intended as examples of how a given Intel-based product, in the specified circumstances and configurations, may affect future costs and provide cost savings. Circumstances will vary. Intel does not guarantee any costs or cost reduction.

Statements in this document that refer to Intel's plans and expectations for the quarter, the year, and the future, are forward-looking statements that involve a number of risks and uncertainties. A detailed discussion of the factors that could affect Intel’s results and plans is included in Intel's SEC filings, including the annual report on Form 10-K.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Performance estimates were obtained prior to implementation of recent software patches and firmware updates intended to address exploits referred to as "Spectre" and "Meltdown." Implementation of these updates may make these results inapplicable to your device or system.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document. Intel does not control or audit third-party benchmark data or the web sites referenced in this document. You should visit the referenced web site and confirm whether referenced data are accurate.

Results have been estimated or simulated using internal Intel analysis or architecture simulation or modeling, and provided to you for informational purposes. Any differences in your system hardware, software or configuration may affect your actual performance.

Intel, the Intel logo, Pentium, Celeron, Atom, Core, Xeon, Movidius, Saffron and others are trademarks of Intel Corporation in the U.S. and/or other countries. *Other names and brands may be claimed as the property of others.

© 2018 Intel Corporation.
<table>
<thead>
<tr>
<th>Library</th>
<th>Commercial Product Version</th>
<th>Opens source version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intel® Math Kernel Library for Deep Neural Networks (Intel® MKL-DNN)</td>
<td>MKL-DNN Github</td>
<td></td>
</tr>
<tr>
<td>Intel® Math Kernel Library (Intel® MKL)</td>
<td>Intel MKL Product site</td>
<td></td>
</tr>
<tr>
<td>Intel® Data Analytics Acceleration Library (Intel® DAAL)</td>
<td>Intel DAAL Product Site</td>
<td>Intel DAAL Github</td>
</tr>
<tr>
<td>Intel® Integrated Performance Primitives (Intel® IPP)</td>
<td>Intel IPP Product Site</td>
<td></td>
</tr>
</tbody>
</table>
# Optimized Frameworks

<table>
<thead>
<tr>
<th>Framework</th>
<th>How to Access Optimized Framework</th>
</tr>
</thead>
<tbody>
<tr>
<td>TensorFlow</td>
<td>Install <a href="https://tensorflow.org">Intel optimized wheel</a> page for CPU optimization instructions</td>
</tr>
<tr>
<td>MXNet</td>
<td>Intel optimizations in main branch via experimental path, <a href="https://mxnet.org">available here</a></td>
</tr>
<tr>
<td>Caffe2</td>
<td>Will upstream to <a href="https://github.com/pytorch/pytorch">master branch</a> in Q2</td>
</tr>
<tr>
<td>PaddlePaddle</td>
<td>Paddle Paddle <a href="https://github.com/PaddlePaddle/PaddlePaddle">master branch</a></td>
</tr>
<tr>
<td>PyTorch</td>
<td>Intel optimizations available in <a href="https://github.com/pytorch/pytorch">this branch</a></td>
</tr>
<tr>
<td>Caffe</td>
<td><a href="https://github.com/pytorch/pytorch">Intel optimized version</a> of Caffe</td>
</tr>
<tr>
<td>CNTK</td>
<td>CNTK <a href="https://github.com/pytorch/pytorch">master branch</a></td>
</tr>
</tbody>
</table>
## Configuration Details 1

<table>
<thead>
<tr>
<th>Benchmark Segment</th>
<th>AI/ML</th>
</tr>
</thead>
<tbody>
<tr>
<td>Benchmark type</td>
<td>Training/Inference</td>
</tr>
<tr>
<td>Benchmark Metric</td>
<td>Images/Sec or Time to train in seconds</td>
</tr>
<tr>
<td>Framework</td>
<td>Caffe</td>
</tr>
<tr>
<td>Topology</td>
<td>1</td>
</tr>
<tr>
<td># of Nodes</td>
<td>1</td>
</tr>
<tr>
<td>Platform</td>
<td>Purley</td>
</tr>
<tr>
<td>Sockets</td>
<td>2S</td>
</tr>
<tr>
<td>Processor</td>
<td>Xeon Platinum, 205W, 28 core, 2.5 GHz</td>
</tr>
<tr>
<td>BIOS</td>
<td>SESC620.868.01.00.00470.040720170855</td>
</tr>
<tr>
<td>Enabled Cores</td>
<td>28</td>
</tr>
<tr>
<td>Platform</td>
<td></td>
</tr>
<tr>
<td>Total Memory</td>
<td>192GB (96 GB per socket)</td>
</tr>
<tr>
<td>Memory Configuration</td>
<td>6 slots/16 GB/2666 Mt/s DDR4 RDIMMs</td>
</tr>
<tr>
<td>Memory Comments</td>
<td>Micron (18ASF2G72PDZ-2G6D1)</td>
</tr>
<tr>
<td>SSD</td>
<td>INTEL SSDSC2KW48 480 GB</td>
</tr>
<tr>
<td>OS</td>
<td>RHEL Server Release 7.2 (Maipo), Linux kernel 3.10.0-327.el7.x86_64</td>
</tr>
<tr>
<td>OS Kernel Comments</td>
<td></td>
</tr>
<tr>
<td>Other Configurations</td>
<td></td>
</tr>
<tr>
<td>HT</td>
<td>ON</td>
</tr>
<tr>
<td>Turbo</td>
<td>ON</td>
</tr>
<tr>
<td>Computer Type</td>
<td>Server</td>
</tr>
<tr>
<td>Framework Version</td>
<td><a href="http://github.com/intel/caffe/">http://github.com/intel/caffe/</a> c7ed32772affa1d9951e2a93d98ed22ab81b88 (release_1.0.6)</td>
</tr>
<tr>
<td>Topology Version, BATCH_SIZE</td>
<td>best (reset -- 50, gnet_v3 -- 224, ssd -- 224)</td>
</tr>
<tr>
<td>Performance command</td>
<td>Inference measured with &quot;caffe time --forward_only --phase TEST&quot; command, training measured with &quot;caffe train&quot; command.</td>
</tr>
<tr>
<td>Data setup</td>
<td>DummyData layer (generating dummy images on the fly)</td>
</tr>
<tr>
<td>Compiler</td>
<td>Intel C++ compiler ver. 17.0.5 20171101</td>
</tr>
<tr>
<td>MKL Library version</td>
<td>intel MKL small libraries version 2018.0.1 20171007</td>
</tr>
<tr>
<td>MKL DNN Library Version</td>
<td></td>
</tr>
<tr>
<td>Performance Measurement Knobs</td>
<td>Environment variables: KMP_AFFINITY=granularity=fine,compact,1.0, OMP_NUM_THREADS=28, CPU Freq set with cpupower frequency-set -d 2.5G -u 3.8G -g performance</td>
</tr>
<tr>
<td>Memory knobs</td>
<td>Caffe run with &quot;numactl -i&quot;.</td>
</tr>
</tbody>
</table>

Performance estimates were obtained prior to implementation of recent software patches and firmware updates intended to address exploits referred to as "Spectre" and "Meltdown." Implementation of these updates may make these results inapplicable to your device or system.
Skylake AI Configuration Details as of July 11th, 2017

Platform: 2S Intel® Xeon® Platinum 8180 CPU @ 2.50GHz (28 cores), HT disabled, turbo disabled, scaling governor set to "performance" via intel_pstate driver, 384GB DDR4-2666 ECC RAM. CentOS Linux release 7.3.1611 (Core), Linux kernel 3.10.0-514.10.2.el7.x86_64. SSD: Intel® SSD DC S3700 Series (800GB, 2.5in SATA 6Gb/s, 25nm, MLC).

Performance measured with: Environment variables: KMP_AFFINITY='granularity=fine, compact', OMP_NUM_THREADS=56, CPU Freq set with cpupower frequency-set -d 2.5G -u 3.8G -g performance

Deep Learning Frameworks:

- Caffe: [link](http://github.com/intel/caffe/), revision f96b759f71b2281835f690af267158b82b150b5c. Inference measured with "caffe time --forward_only" command, training measured with "caffe time" command. For "ConvNet" topologies, dummy dataset was used. For other topologies, data was stored on local storage and cached in memory before training. Topology specs from [https://github.com/intel/caffe/tree/master/models/intel_optimized_models](https://github.com/intel/caffe/tree/master/models/intel_optimized_models) (GoogLeNet, AlexNet, and ResNet-50), [https://github.com/intel/caffe/tree/master/models/default_vgg_19](https://github.com/intel/caffe/tree/master/models/default_vgg_19) (VGG-19), and [https://github.com/soumith/convnet-benchmarks/tree/master/caffe/imagenet_winners](https://github.com/soumith/convnet-benchmarks/tree/master/caffe/imagenet_winners) (ConvNet benchmarks; files were updated to use newer Caffe prototxt format but are functionally equivalent). Intel C++ compiler ver. 17.0.2 20170213, Intel MKL small libraries version 2018.0.20170425. Caffe run with "numactl -l".

- TensorFlow: [https://github.com/tensorflow/tensorflow](https://github.com/tensorflow/tensorflow), commit id 207203253b6f8ea5e938a512798429f91d5b4e7e. Performance numbers were obtained for three convnet benchmarks: alexnet, googlenetv1, vgg[https://github.com/soumith/convnet-benchmarks/tree/master/master/tensorflow](https://github.com/soumith/convnet-benchmarks/tree/master/master/tensorflow) using dummy data. GCC 4.8.5, Intel MKL small libraries version 2018.0.20170425, interop parallelism threads set to 1 for alexnet, vgg benchmarks, intra op parallelism threads set to 56, data format used is NCHW, KMP_BLOCKTIME set to 1 for googlenet and vgg benchmarks, 30 for the alexnet benchmark. Inference measured with --caffe time -forward_only -engine MKL2017option, training measured with --forward_backward_only option.

- MxNet: [https://github.com/dmlc/mxnet/](https://github.com/dmlc/mxnet/), revision 56fd91a71f36fea483e882b0358c8d465b54a20. Dummy data was used. Inference was measured with "benchmark_score.py", training was measured with a modified version of benchmark_score.py which also runs backward propagation. Topology specs from [https://github.com/dmlc/mxnet/tree/master/example/image-classification/symbols](https://github.com/dmlc/mxnet/tree/master/example/image-classification/symbols). GCC 4.8.5, Intel MKL small libraries version 2018.0.20170425.

- Neon: ZP/MKL_CHWN branch commit id:52bd02ac8a947a2adabb8a227166a7da5d9123b6d. Dummy data was used. The main.py script was used for benchmarking, in mkl mode. ICC version used : 17.0.3 20170404, Intel MKL small libraries version 2018.0.20170425.

Performance estimates were obtained prior to implementation of recent software patches and firmware updates intended to address exploits referred to as "Spectre" and "Meltdown." Implementation of these updates may make these results inapplicable to your device or system.
Broadwell AI Configuration Details as of July 11th, 2017

Platform: 2S Intel® Xeon® CPU E5-2699 v4 @ 2.20GHz (22 cores), HT enabled, turbo disabled, scaling governor set to “performance” via acpi-cpufreq driver, 256GB DDR4-2133 ECC RAM. CentOS Linux release 7.3.1611 (Core), Linux kernel 3.10.0-514.10.2.el7.x86_64. SSD: Intel® SSD DC S3500 Series (480GB, 2.5in SATA 6Gb/s, 20nm, MLC).

Performance measured with: Environment variables: KMP_AFFINITY='granularity=fine, compact,1,0', OMP_NUM_THREADS=44, CPU Freq set with cpupower frequency-set -d 2.2G -u 2.2G -g performance

Deep Learning Frameworks:

- **Caffe** ([http://github.com/intel/caffe/](http://github.com/intel/caffe/)), revision f96b759f71b2281835f690af267158b82b150b5c. Inference measured with “caffe time --forward_only” command, training measured with “caffe time” command. For “ConvNet” topologies, dummy dataset was used. For other topologies, data was stored on local storage and cached in memory before training. Topology specs from [https://github.com/intel/caffe/tree/master/models/intel_optimized_models](https://github.com/intel/caffe/tree/master/models/intel_optimized_models) (GoogLeNet, AlexNet, and ResNet-50), [https://github.com/intel/caffe/tree/master/models/default_vgg_19](https://github.com/intel/caffe/tree/master/models/default_vgg_19) (VGG-19), and [https://github.com/soumith/convnet-benchmarks/tree/master/caffe/imagenet_winners](https://github.com/soumith/convnet-benchmarks/tree/master/caffe/imagenet_winners) (ConvNet benchmarks; files were updated to use newer Caffe prototxt format but are functionally equivalent). GCC 4.8.5, Intel MKL small libraries version 2017.0.2.20170110.

- **TensorFlow** ([https://github.com/tensorflow/tensorflow](https://github.com/tensorflow/tensorflow)), commit id 207203253b6f8ea5e938a512798429f91d5b4e7e. Performance numbers were obtained for three convnet benchmarks: alexnet, googlenetv1, vgg([https://github.com/soumith/convnet-benchmarks/tree/master/tensorflow](https://github.com/soumith/convnet-benchmarks/tree/master/tensorflow)) using dummy data. GCC 4.8.5, Intel MKL small libraries version 2018.0.20170425, interop parallelism threads set to 1 for alexnet, vgg benchmarks, 2 for googlenet benchmarks, interop parallelism threads set to 44, data format used is NCHW, KMP_BLOCKTIME set to 1 for googlenet and vgg benchmarks, 30 for the alexnet benchmark. Inference measured with --caffe time --forward_only --engine MKL2017option, training measured with --forward_backward_only option.

- **MxNet** ([https://github.com/dmlc/mxnet/](https://github.com/dmlc/mxnet/)), revision e9f281a27584cdb78a6b6e648b3dbcb0d3d7. Dummy data was used. Inference was measured with “benchmark_score.py”, training was measured with a modified version of benchmark_score.py which also runs backward propagation. Topology specs from [https://github.com/dmlc/mxnet/tree/master/example/image-classification/symbols](https://github.com/dmlc/mxnet/tree/master/example/image-classification/symbols). GCC 4.8.5, Intel MKL small libraries version 2017.0.2.20170110.

- **Neon** ZP/MKL_CHWN branch commit id:52bd02ac947a2adabb8a227166a7da5d9123b6d. Dummy data was used. The main.py script was used for benchmarking, in mkl mode. ICC version used : 17.0.3 20170404, Intel MKL small libraries version 2018.0.20170425.

Performance estimates were obtained prior to implementation of recent software patches and firmware updates intended to address exploits referred to as "Spectre" and "Meltdown." Implementation of these updates may make these results inapplicable to your device or system.
<table>
<thead>
<tr>
<th>Configuration Details May8th 2018</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Benchmark Segment</strong></td>
</tr>
<tr>
<td><strong>Benchmark Type</strong></td>
</tr>
<tr>
<td><strong>Benchmark Metric</strong></td>
</tr>
<tr>
<td><strong>Framework</strong></td>
</tr>
<tr>
<td><strong>Topology</strong></td>
</tr>
<tr>
<td><strong># of Nodes</strong></td>
</tr>
<tr>
<td><strong>Platform</strong></td>
</tr>
<tr>
<td><strong>Sockets</strong></td>
</tr>
<tr>
<td><strong>Processor</strong></td>
</tr>
<tr>
<td><strong>BIOS</strong></td>
</tr>
<tr>
<td><strong>Enabled Cores</strong></td>
</tr>
<tr>
<td><strong>Slots</strong></td>
</tr>
<tr>
<td><strong>Total Memory</strong></td>
</tr>
<tr>
<td><strong>Memory Configuration</strong></td>
</tr>
<tr>
<td><strong>Memory Comments</strong></td>
</tr>
<tr>
<td><strong>SSD</strong></td>
</tr>
<tr>
<td><strong>OS</strong></td>
</tr>
<tr>
<td><strong>OS/Kernel Comments</strong></td>
</tr>
<tr>
<td><strong>Other Configurations</strong></td>
</tr>
<tr>
<td><strong>HT</strong></td>
</tr>
<tr>
<td><strong>Turbo</strong></td>
</tr>
<tr>
<td><strong>Computer Type</strong></td>
</tr>
<tr>
<td><strong>Topology Version, BATCHSIZE</strong></td>
</tr>
<tr>
<td><strong>Dataset, version</strong></td>
</tr>
<tr>
<td><strong>Performance command</strong></td>
</tr>
<tr>
<td><strong>Data setup</strong></td>
</tr>
<tr>
<td><strong>Compiler</strong></td>
</tr>
<tr>
<td><strong>Performance Measurement Knobs</strong></td>
</tr>
<tr>
<td><strong>Memory knobs</strong></td>
</tr>
<tr>
<td><strong>caffe branch:</strong> origin/master</td>
</tr>
<tr>
<td><strong>MKLDNN version:</strong></td>
</tr>
</tbody>
</table>

Performance estimates were obtained prior to implementation of recent software patches and firmware updates intended to address exploits referred to as "Spectre" and "Meltdown". Implementation of these updates may make these results inapplicable to your device or system.
## Configuration Details of Amazon EC2 C5.18xlarge 1 Node Systems

<table>
<thead>
<tr>
<th>Configuration Details</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Benchmark Segment</td>
<td>AI/ML</td>
</tr>
<tr>
<td>Benchmark type</td>
<td>Inference</td>
</tr>
<tr>
<td>Benchmark Metric</td>
<td>Sentence/Sec</td>
</tr>
<tr>
<td>Framework</td>
<td>Official mxnet</td>
</tr>
<tr>
<td>Topology</td>
<td>GNMT(sockeye)</td>
</tr>
<tr>
<td># of Nodes</td>
<td>1</td>
</tr>
<tr>
<td>Platform</td>
<td>Amazon EC2 C5.18xlarge instance</td>
</tr>
<tr>
<td>Sockets</td>
<td>2S</td>
</tr>
<tr>
<td>Processor</td>
<td>Intel® Xeon® Platinum 8124M CPU @ 3.00GHz (Skylake)</td>
</tr>
<tr>
<td>BIOS</td>
<td>N/A</td>
</tr>
<tr>
<td>Enabled Cores</td>
<td>18 cores / socket</td>
</tr>
<tr>
<td>Platform</td>
<td>N/A</td>
</tr>
<tr>
<td>Slots</td>
<td>N/A</td>
</tr>
<tr>
<td>Total Memory</td>
<td>144GB</td>
</tr>
<tr>
<td>Memory Configuration</td>
<td>N/A</td>
</tr>
<tr>
<td>SSD</td>
<td>EBS Optimized 200GB, Provisioned IOPS SSD</td>
</tr>
<tr>
<td>OS</td>
<td>Red Hat 7.2 (HVM) Amazon Elastic Network Adapter (ENA) Up to 10 Gbps of aggregate network bandwidth</td>
</tr>
<tr>
<td>Network Configurations</td>
<td>Installed Enhanced Networking with ENA on Centos Placed the all instances in the same placement</td>
</tr>
<tr>
<td>HT</td>
<td>ON</td>
</tr>
<tr>
<td>Turbo</td>
<td>ON</td>
</tr>
<tr>
<td>Computer Type</td>
<td>Server</td>
</tr>
</tbody>
</table>

Config 34
## Configuration details of Amazon EC2 C5.18xlarge 1 node systems

<table>
<thead>
<tr>
<th>Framework Version</th>
<th>mxnet mkldnn: <a href="https://github.com/apache/incubator-mxnet/">https://github.com/apache/incubator-mxnet/</a> 4950f6649e329b23a1efdc40aaa25260d47b4195</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topology Version</td>
<td>GNMT: <a href="https://github.com/awslabs/sockeye/tree/master/tutorials/wmt">https://github.com/awslabs/sockeye/tree/master/tutorials/wmt</a></td>
</tr>
<tr>
<td>Batch size</td>
<td>GNMT: 1 2 8 16 32 64 128</td>
</tr>
<tr>
<td>MKLDNN</td>
<td>F5218ff4fd2d16d13aada2e632afdf2514fee3</td>
</tr>
<tr>
<td>Compiler</td>
<td>g++: 4.8.5&lt;br&gt;gcc: 7.2.1</td>
</tr>
</tbody>
</table>
# Dawnbench Configurations

<table>
<thead>
<tr>
<th>EC2 Instance type</th>
<th>Machine Type</th>
<th>vCPU (#s)</th>
<th>Memory (GiB)</th>
<th>Disk(GB)</th>
<th>Storage (mbps, EBS bandwidth)</th>
<th>Ethernet (Gigabit)</th>
<th>Price ($ per Hour)</th>
</tr>
</thead>
<tbody>
<tr>
<td>C5.2xlarge</td>
<td>SKX 8124M</td>
<td>8</td>
<td>16</td>
<td>128</td>
<td>Up to 2,250</td>
<td>Up to 10</td>
<td>0.34</td>
</tr>
<tr>
<td></td>
<td>4 cores 3 GHz base frequency</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C5.4xlarge</td>
<td>SKX 8124M</td>
<td>16</td>
<td>32</td>
<td>128</td>
<td>2,250</td>
<td>Up to 10</td>
<td>0.68</td>
</tr>
<tr>
<td></td>
<td>8 cores 3 GHz base frequency</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C5.18xlarge</td>
<td>SKX 8124M</td>
<td>72</td>
<td>144</td>
<td>128</td>
<td>9,000</td>
<td>25</td>
<td>3.06</td>
</tr>
<tr>
<td></td>
<td>2S x 18 cores 3 GHz base frequency</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Dawnbench IntelCaffe Inference topology

Inference: INT8 Resnet50 with 15% pruning, 53% performance gain over FP32

- Based on 93.3% accuracy FP32 Resnet50 topology
- Pure INT8 except for first convolution layer: <0.2% accuracy drop, 43% performance gain over FP32
- 15% filter pruned according to KL distance: <0.1% accuracy drop, 15% performance gain over FP32
Config 10

Intel® and SURFsara* Research Collaboration
MareNostrum4/BSC* Configuration Details

*MareNostrum4/Barcelona Supercomputing Center: [https://www.bsc.es/](https://www.bsc.es/)

**Compute Nodes**: 2 sockets Intel® Xeon® Platinum 8160 CPU with 24 cores each @ 2.10GHz for a total of 48 cores per node, 2 Threads per core, L1d 32K; L1i cache 32K; L2 cache 1024K; L3 cache 33792K, 96 GB of DDR4, Intel® Omni-Path Host Interface, dual-rail. Software: Intel® MPI Library 2017 Update 4Intel® MPI Library 2019 Technical Preview OFI 1.5.OSM2 w/ Multi-EF, 10 Gbit Ethernet, 200 GB local SSD, Red Hat® Enterprise Linux 6.7.

**Intel® Caffe**: Intel® version of Caffe; [http://github.com/intel/caffe/](http://github.com/intel/caffe/), revision 80127bf2bf70231cbe7f5de0b1bc11de4a69.


**Time-To-Train**: measured using "train" command. Data copied to memory on all nodes in the cluster before training. No input image data transferred over the fabric while training;

**Performance measured with**: export OMP_NUM_THREADS=44 (the remaining 4 cores are used for driving communication), export I_MPI_FABRICS=tmi, export I_MPI_TMI_PROVIDER=psm2

OMP_NUM_THREADS=44 KMP_AFFINITY="proclist=[0-87],granularity=thread,explicit" KMP_HW_SUBSET=1t MLSL_NUM_SERVERS=4 mpiexec.hydra -l -n
$SLURM_JOB_NUM_NODES -ppn 1 -f hosts2 -genv OMP_NUM_THREADS 44 -env KMP_AFFINITY="proclist=[0-87],granularity=thread,explicit" -env KMP_HW_SUBSET 1t -genv I_MPI_FABRICS tmi -genv I_MPI_HYDRA_BRANCH_COUNT $SLURM_JOB_NUM_NODES -genv I_MPI_HYDRA_PMI_CONNECT alltoall sh -c cat/ilsvrc12_train_lmdb_striped_64/data.mdb > /dev/null; cat/ilsvrc12_val_lmdb_striped_64/data.mdb > /dev/null; ulimit -u 8192; ulimit -a; numactl -H; /caffe/build/tools/caffe train -solver=/caffe/models/intel_optimized_models/multinode/resnet_50_256_nodes_8k_batch/solver_poly_quick_large.prototxt -engine "MKL2017"


*SURFsara B.V. is the Dutch national high-performance computing and e-Science support center. Amsterdam Science Park, Amsterdam, The Netherlands.

---

Performance estimates were obtained prior to implementation of recent software patches and firmware updates intended to address exploits referred to as "Spectre" and "Meltdown." Implementation of these updates may make these results inapplicable to your device or system.
Stampede2*/TACC* Configuration Details

*Stampede2/TACC: https://portal.tacc.utexas.edu/user-guides/stampede2

Compute Nodes: 2 sockets Intel® Xeon® Platinum 8160 CPU with 24 cores each @ 2.10GHz for a total of 48 cores per node, 2 Threads per core, L1d 32K; L1i cache 32K; L2 cache 1024K; L3 cache 33792K, 96 GB of DDR4, Intel® Omni-Path Host Fabric Interface, dual-rail. Software: Intel® MPI Library 2017 Update 4Intel® MPI Library 2019 Technical Preview OFI 1.5.0PSM2 w/ Multi-EP, 10 Gbit Ethernet, 200 GB local SSD, Red Hat* Enterprise Linux 6.7.


Model: Topology specs from https://github.com/intel/caffe/tree/master/models/intel_optimized_models (ResNet-50) and modified for wide-RedNet-50.; Batch size as stated in the performance chart

Performance measured with:
export OMP_NUM_THREADS=10 Per Worker (the remaining 2 cores are used for driving communication), export I_MPI_FABRICS=tmi, export I_MPI_TMI_PROVIDER=psm2

OMP_NUM_THREADS=10 KMP AFFINITY="proc[0-63],granularity=thread,explicit" KMP_HW_SUBSET=1t MLSL_NUM_SERVERS=4 mpiexec.hydra -PSM2 -l -n $SLURM JOB NUM NODES -ppn 1 -f hosts2 -genv OMP_NUM_THREADS 64 -genv KMP AFFINITY "proc[0-63],granularity=thread,explicit" -env KMP_HW_SUBSET 1t -genv I_MPI_FABRICS tmi -genv I_MPI HYDRA BRANCH COUNT $SLURM JOB NUM NODES -genv I_MPI HYDRA PMI CONNECT alltoall sh -c 'cat /ilsrrc12_train_lmdb striped_64/data.mdb > /dev/null ; cat /ilsrrc12_val_lmdb striped_64/data.mdb > /dev/null ; ulimit -u 8192 ; ulimit -a ; numactl -H ; /caffe/build/tools/caffe train -- solver=/caffe/models/intel_optimized_models/multinode/resnet_50_256_nodes_8k_batch/solver_poly_quick_large.prototxt -engine "MKL2017"

Configuration Details on Slide: VLAB at Intel® Configuration Details:
Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit: http://www.intel.com/performance. Copyright © 2017, Intel Corporation
Case Study: Time-series Pattern Detection
Leading U.S. Market Exchange

**Client:** Leading U.S. market exchange

**Challenge:** Identify known patterns or anomalies in market trading data in order to predict investment activity, such as fraudulent activity or spikes in trading volume and whether any action is required.

**Solution:** Using Intel® Nervana™ Cloud and Neon framework. Built a recurrent neural network (RNN)-based model, utilizing encoders and decoders, in order to ingest public order book data and automatically learn informative patterns of activity in the historical data. Time series analysis enables new use cases for fraud detection, anomaly detection, and other future applications.

**Result**

10X **Reduction**

In data storage and search complexity costs, with more accurate matches than non-deep learning approach.
PROOF OF CONCEPT: IMAGE RECOGNITION

SEISMIC REFLECTION ANALYSIS

- Client:
  - A leading developer of software solutions to the global oil and gas industry.
- Challenge:
  - Automate identification of fault lines within seismic reflection data.
- Solution:
  - Built a proof of concept that is trained using seismic reflection data and can predict the probability of finding fault lines on previously unseen images.
  - Performs pixel-wise semantic segmentation of SEG-Y formatted data
  - Model trained using supervised learning
- Advantages:
  - Automation enables analysis of vast amounts of data faster
  - Could identify potentially rewarding locations from subtle clues in the data
CASE STUDY: ENTERPRISE ANALYTICS

SERPRO*

RESULT
$1 BILLION

Streamlined collection of US $1 billion in revenue by designing new APIs for car and license plate image recognition.

Client: SERPRO, Brazil's largest government-owned IT services corporation, providing technology services to multiple public sector agencies.

Challenge: Across Brazil, 35,000 traffic enforcement cameras document 45 million violations every year, generating US $1 billion in revenue. Fully automating the complex, labor-intensive process for issuing tickets by integrating image recognition via AI could reduce costs and processing time.

Solution: Used deep learning techniques to optimize SERPRO's code. With Brazilian student-partners, developed new algorithms, training and inference tests using Google TensorFlow* on Dell EMC PowerEdge R740*, running on Intel® Xeon® Scalable processor-based platforms.

*Other names and brands may be claimed as the property of others.